

Validating the Restricted Focus Viewer: A study using eye-movement tracking

ROMAN BEDNARIK AND MARKKU TUKIAINEN
University of Joensuu, Joensuu, Finland

Investigation of cognitive processes and visual attention during problem-solving tasks is an important part of understanding human reasoning. Eyetracking technology has proven to have many benefits in revealing visual attention patterns. However, the high price of accurate eyetrackers and the difficulties associated with using them represent major obstacles to their wider application. Therefore, previous studies have sought to find alternatives to eyetracking. The Restricted Focus Viewer (RFV) brings a small part of an otherwise blurred display to the focus of visual attention: A user controls what part of the screen is in focus by using a computer mouse and explicitly selecting the area to be shown in focus. Recently, some studies have employed the RFV to investigate cognitive behavior of users, and some researchers have even enhanced the tool to study usability. We replicated a previous RFV-based study while also recording gaze data. We compared the attention allocation in time and space as reported by the RFV and an eyetracker. Further, we investigated the effects of RFV's display blurring on the visual attention allocation of 18 novice and expert programmers. Our results indicate that the data obtained from the two tools differ. Also, the RFV-blurring interferes with the strategies utilized by experts, and has an effect on fixation duration. However, task performance was preserved.

Researchers who investigate cognitive processing during reasoning tasks have several options for how to get insights into the behavior and strategies exhibited by the participants. Especially in situations in which the stimulus is visual and the reasoning is related (and dependent on it), eyetracking systems have proven useful in revealing the patterns of visual attention during the task. Typical examples of successful applications of eye-movement tracking include studies relating eye-movement patterns to cognitive processes (Just & Carpenter, 1976), studies on reading (e.g., Rayner, 1998), and studies that investigated differences between novices and experts with regard to eye-movement patterns (e.g., Hyönä, Lorch, & Kaakinen, 2002; Law, Atkins, Kirkpatrick, & Lomax, 2004). However, the relatively high price of accurate eyetracking equipment, and other issues (such as drift, a need for calibration, and a certain level of obtrusiveness), have prevented a wider application. To remedy some of the problems and limitations typical of eyetracking, researchers have sought to develop cheap, yet accurate, alternatives.

Recently, the Restricted Focus Viewer (RFV; Jansen, Blackwell, & Marriott, 2003), a tool that blurs the display and restricts users to only a small focused point, has been introduced as an alternative tool to track visual attention, and it has also been used in various studies. The RFV has been validated in two experiments run by Jansen et al. (2003); however, these validations involved only relatively simple reasoning with visual stimuli. In other studies, the RFV-based technology has been applied (1) to discover

strategies of participants debugging computer programs, with the aid of multiple and linked visual representations of the programs (Romero, Cox, du Boulay, & Lutz, 2002; Romero, du Boulay, Cox, & Lutz, 2003; Romero, du Boulay, Lutz, & Cox, 2003; Romero, Lutz, Cox, & du Boulay, 2002), (2) to investigate the issues of usability of hyper-linked documents (Tarasewich & Fillion, 2004), and (3) in a study of shifts of visual attention during integration of text and graphics (Futrelle & Rumshisky, 2001). For the purposes of visual attention tracking during complex reasoning tasks such as computer program debugging, the use of the RFV-based approach has been questioned (Bednarik & Tukiainen, 2004).

In this article, we present a replication of a previous study by Romero and colleagues, in which visual attention of computer programmers was recorded using the RFV. In our study, together with the RFV-based measurement of visual attention, we used a remote eyetracker to collect the patterns of eye movements of participants who were debugging computer programs. Although the RFV has already been validated in previous studies, this validation has involved only relatively simple tasks. It remains an open question whether the behavior of experimental participants during cognitively more demanding tasks is influenced by having to manually move the only focused spot within otherwise blurred visual stimuli.

This article contributes to the body of knowledge in many ways. First, it is important in validating and cross-validating the tools employed in the experiments investi-

gating visual attention and cognitive processing in general. Although the RFV technology is not supposed to replace eye-movement-tracking systems, it aims to provide accurate information about visual attention allocation during reasoning with visual stimuli. By replicating one of the previous studies in which the RFV was employed, we compare the RFV-based measurement with the data collected using a remote eye-movement tracker. Second, we investigate a possible effect of the display-blurring technique on the behavior and performance of experimental participants. Recently, some researchers have started to use the RFV for recording visual attention patterns during tasks involving complex reasoning. We provide important information about (1) how the actual implementation of the RFV technique influences the participants, (2) the data collected, and therefore also (3) the conclusions drawn from the experiments.

Previous Related Studies

The RFV (Jansen et al., 2003) has been developed as an alternative to eye-movement-tracking technologies. One of the main claimed advantages of RFV is that it allows for an automated collection of participants' foci of visual attention. The RFV blurs the stimulus image and displays it on a computer screen, allowing the participant to see only a limited focused region at a time. In order to have another part of the stimuli focused, the participant has to move the focused region using the computer mouse. The RFV then records the moves over the stimuli, which are stored for later analysis. The tool collects time-stamped data about mouse and keyboard events, the currently focused region's index, the total durations of sessions, and other events. Voice protocols can be recorded along with the interaction data. The RFV is of course not capable of collecting visual attention data when the blurring is turned off.

Analysis and interpretation of the data recorded by RFV, and the data's relation to the investigated task, are (as is also true for eyetracking) up to the researcher. Usually, the so-called areas of interest (AOIs) are defined within the interface, and several metrics can be computed—for example, the total or proportional time spent on an AOI.

In the context of Java program debugging with multiple representations, a modified version of the RFV has been employed in various studies (Romero, Cox, et al., 2002; Romero, du Boulay, Cox, & Lutz, 2003; Romero, du Boulay, Lutz, & Cox, 2003; Romero, Lutz, et al., 2002). In these studies, a software development environment (SDE) was built on top of the RFV and was employed to track visual attention to investigate the coordination strategies of programmers debugging Java programs, who were working with multiple adjacent representations. Several eye-movement-like metrics were derived, to identify superior debugging strategies of participants or to measure the effects of different visualizations on the coordination strategies. For example, a balance in accumulated fixation times between different representations (regions) could reflect a good debugging performance (Romero, du Boulay, Lutz, & Cox, 2003). Another measure derived in the aforementioned studies was the number of switches per

minute between the representations of a program. More experienced programmers were found to switch more frequently between the main representations.

Previous research focusing on program debugging and differences between novice and skilled programmers has shown the superiority of expert programmers over novices in terms of domain knowledge, performance, and strategies: Expert programmers found more bugs, found them faster, and tended to spend more time on building a mental model of the problem (Gugerty & Olson, 1986). Experts also (1) are more able to remember specific parts of the source code (Fix, Wiedenbeck, & Scholtz, 1993), (2) focus only on relevant information needed to solve the problem (Koenemann & Robertson, 1991), (3) are not committed to one interpretation, as novices are (Vessey, 1985), and (4) are therefore able to change their strategies as needed.

Figure 1 shows a screen shot of the SDE when the restricted condition of the RFV was turned on. The SDE's interface contains three main areas: The code is in the left window pane, the frame containing a visualization of the program is in the top-right pane, and the output is at the bottom right. In Figure 1, the focused region, as set by the user's mouse, is located over the visualization in the top-right pane.

The RFV has also been employed in the research of shifts of visual attention during integration of text and graphics (Futrelle & Rumshisky, 2001). Because the original version of the RFV was limited to only static stimuli, other researchers tried to improve the RFV idea, using an enhanced version of RFV for usability studies of hyperlinked documents (Tarasewich & Fillion, 2004).

Experiment

We are very interested in the possibilities and limitations of experimental tools, and want to find out whether their use actually interferes with the (otherwise unaltered) behavior of the participants in an experiment. The purposes of the present experiment were (1) to compare the RFV-based measurements of visual attention shifts to the data obtained through eye-movement tracking, and (2) to investigate possible interference of the blurring technique used by the RFV with the debugging strategies and performance. We fully replicated a previous experiment (Romero, Lutz, et al., 2002) that employed the RFV; a remote eye-movement tracker also recorded the visual attention shifts of the participants. The changes in visual attention when the stimuli were blurred and fully focused were analyzed.

METHOD

Participants

A total of 19 participants were recruited from a population of students, researchers, and teachers from the authors' department. One participant withdrew from the experiment prematurely; therefore, the analysis is based on the data recorded from 18 participants. All of the participants had normal or corrected-to-normal vision, according to their own report, and had never taken part in an eyetracking experiment. The average age was 25.3 ($SD = 4.4$) years. Three

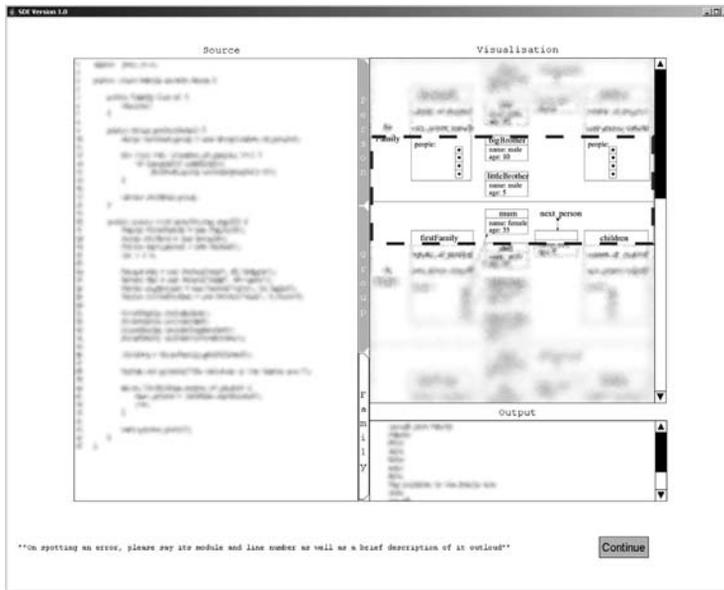


Figure 1. The software debugging environment built on top of the RFV. The focused area is at the top right on the visualization (dashed line added by the authors).

participants were female. The programming and Java experience of the participants varied: Some had just passed a Java course and had little experience, whereas others were professionals working in programming-related careers. Experience with Java programming, and especially professional background, was chosen as the main factor for classification. The participants were divided into two groups. The less experienced group consisted of 10 programmers, who had an average of 63 ($SD = 33.1$) months of programming experience, 8.13 ($SD = 6.1$) months of which were Java programming. No novice participant had ever worked as a professional programmer. The expert group was formed from the remaining 8 participants; they had a mean programming experience of 96 ($SD = 24.7$) months and a mean Java experience of 16.25 ($SD = 17.9$) months, and all of them, except 1, had professional programming experience.

Materials and Design

The target programs used in this study were identical to those used in Romero, Lutz, et al. (2002). The warm-up program inspected whether a point was inside a rectangle. The first program (the “Family” program) printed out the names of the children of a sample family, and the second program (the “Till” program) counted the cash in a cash-register till, giving subtotals for the different denominations of coins. In their study, Romero, Lutz, et al. (2002) had two versions of the target programs; the main difference between the versions was that the second one was a more sophisticated version of the first one. In our replication, we used the less sophisticated versions of Romero’s programs and the graphical functional representations in visualizations.

The two main experimental programs contained four errors each; the warm-up program was seeded with two errors. Following the classification of the errors established in Romero, Cox, et al. (2002), Romero, du Boulay, Cox, and Lutz (2003), Romero, du Boulay, Lutz, and Cox (2003), and Romero, Lutz, et al. (2002), the errors in the target programs could be classified as functional, control-flow, and data-structure errors. There were no syntactical errors in the programs (all of the programs could be compiled), and the participants were notified of this.

We used a mixed design with one within-subjects factor (RFV restricting condition, hereafter referred to as RFV-on/RFV-off) and

one between-subjects factor (level of experience), with four dependent variables (number of errors spotted, accumulated fixation time, mean fixation duration, and switching frequency, as measured by an eyetracker). The accumulated fixation time is the total time a participant spent during a session fixating an area of interest (AOI). For an AOI, all of the fixations were summed, and the number was divided by the total fixation count throughout the debugging session, giving the mean fixation duration. The switching frequency refers to the average number of attention switches per minute between each of the AOIs. The mean fixation duration is a measure related to the depth of required processing and therefore to the mental workload during the task (Goldberg & Kotval, 1998). Most of the results were analyzed by performing ANOVA and/or planned paired t tests.

Procedure

The experiment was conducted in a quiet laboratory. Each participant was seated in a comfortable chair facing a 17-in. TFT display, at a viewing distance of about 80 cm. Before the experiment, the participants had to successfully go through an automatic eyetracking calibration procedure. After that, the participants read detailed instructions about the experiment and the environment used. Participants debugged three programs. The first warm-up session was performed under the RFV restricted-view (RFV-on) condition so that the participants could become familiar with controlling the focused spot and operating the debugging environment. Then, the two main debugging sessions were performed; one session was performed under the RFV-on condition, and the other session was performed under the RFV-off condition, in which the whole display was presented in focus. The order of the programs and conditions was counterbalanced.

Each session had two phases. First, the specification of the program was displayed. It described the problem the program was supposed to solve and the approach to the solution. Two sample interactions were provided—the desired behavior and the actual behavior of the program. Second, the participants were given 10 min (this limitation was taken from the previous studies) to debug the program; they were instructed to find as many errors as possible and to report them aloud.

Finally, after the debugging session, the participants were informally interviewed.

Apparatus

The SDE used in the previous studies (Romero, Cox, et al., 2002; Romero, du Boulay, Cox, & Lutz, 2003; Romero, du Boulay, Lutz, & Cox, 2003; Romero, Lutz, et al., 2002) was employed for the experiment as a source of stimuli. In these studies and in the present experiment, RFV Version 2.1 was used. The settings of the underlying RFV-based mechanism for blurring the stimuli were the same as in the replicated studies (e.g., Romero, Cox, et al., 2002); the program code, the visualization, and the output were precomputed and static. There were three transition steps between the clear view and the fully blurred stimuli, and all of the regions were rectangular, with widths and heights as follows (from outermost to the fully focused region, in pixels): 800 × 400, 700 × 300, 620 × 240, and 540 × 190. It was not possible to read the source code when it was blurred, and the speed of motion blur was set to 1 pixel per sec. Figure 2 illustrates the level of RFV-blurring in the experiment.

For eyetracking, the remote Tobii ET-1750 (sampling rate of 30 Hz) eyetracker was used. The device is built into the 17-in. display (resolution of 1,280 × 1,024), makes no contact with participants, and contains no movable or audible parts that could possibly interfere with the participants (Figure 3). The eyetracking data were collected throughout the whole experiment; the RFV was able to collect data only in the RFV-on condition. The AOIs were defined to correspond with the three main panels in the SDE window: the code, the visualization, and the output pane.

RESULTS

The RFV as a Visual Attention Tracking Tool

To investigate the ability of the RFV to accurately record the switches in visual attention between the areas of the interface, we compared the number of switches per minute. Six types of switches, between three main areas (code, visualization, and output), were possible, as shown in Figure 4. We compared the switching behavior as measured by the RFV to the number of switches as measured by the eyetracker, and we analyzed the differences with the blurring (RFV-on) and without the blurring (RFV-off). Because the RFV cannot measure any switching in visual attention without having the display blurred, only data from the eyetracker were available for that condition.

Three separate two-way ANOVAs (tool × switch type) were run to compare the conditions and measurement tools. First, the RFV- and eyetracking-based data under the RFV-on condition were compared, to investigate the effects of tool. The main effect of tool on the number of switches was significant [$F(1,17) = 30.23, p < .001$], and the interaction effect between tool and switch type was significant [$F(5,85) = 7.83, p < .001$].

The second comparison included the number of switches as measured by the eyetracker, with and without the blurring condition. The main effect of the blurring on the number of switches was significant [$F(1,17) = 5.90, p < .05$], and the interaction between blurring and type of switch was not significant [$F(5,85) = 0.48, n.s.$].

Finally, to compare the tools in their natural settings, we analyzed the differences between the number of switches as measured by the RFV in the RFV-on condition and the number of switches as measured by the eyetracker in the RFV-off condition. The main effect of tool and condition was significant [$F(1,17) = 32.62, p < .001$], and the interaction effect with type of switch was also significant [$F(5,85) = 5.58, p < .001$]. For all three comparisons, there was a main effect of switch type on the number of switches [$F(5,85) = 35.37, p < .001$; $F(5,85) = 23.94, p < .001$; $F(5,85) = 22.32, p < .001$, respectively].

The shapes of the amplitudes of the values shown in Figure 4 might suggest that the data obtained by the two tools could be systematically correlated. In other words, the RFV might constantly report lower numbers of switches than the eyetracker, and the relation between these two might be linear. To complement the previous results, we therefore analyzed how the measurements from the two tools and between the two blurring conditions were correlated (Table 1). No significant correlation was found between the measures obtained using the RFV and those recorded by the eyetracker.

Effects of Display Blurring

Since the previous results raised some doubts about the reliability of the RFV technique, the rest of the results, related to the behavior of the experimental groups, were obtained using only the remote eyetracker.

Effects of display blurring on debugging performance. The debugging performance was measured by the number of errors spotted. Under the RFV-on condition, the less experienced group found 2.1 ($SD = 1.10$) errors, on average, and the more experienced group spotted 3.125 ($SD = 0.84$) errors, on average [$t(16) = 2.18, p < .05$]. Under the RFV-off condition, the less experienced group found 2.1 ($SD = 0.88$) errors, on average, and the more experienced group spotted 2.88 ($SD = 1.13$) errors, on average [$t(16) = 1.65, n.s.$]. According to an ANOVA, the effect of the restricted-view condition on the debugging performance was not significant, whereas the effect of

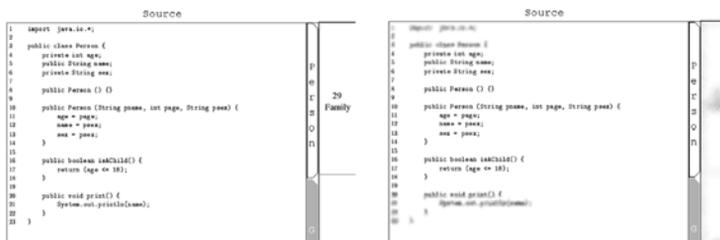


Figure 2. Two snapshots detailing the source panel with RFV blurring off (left) and on (right).



Figure 3. The remote eyetracking device used in the experiment.

experience on the number of bugs found was significant [$F(1,16) = 5.28, p < .05$].

Effects of display blurring on debugging strategies. According to ANOVA, the distribution of relative accumulated fixation time, as measured by the eyetracker, was not affected by the RFV condition for either of the two experimental groups (Figure 5). Novice participants spent, on average, 82% of the total time fixating on the code panel, 14% of the time on visualization, and 4% of the time on the output AOI. For experts, the relative accumulated fixation time followed a distribution of 87%, 10%, and 3%, respectively. Although no effect of experience on the fixation time was found, the effect of AOI was significant [$F(2,32) = 623.07, p < .001$].

Dynamics of attention-switching behavior were measured by the eyetracker as the number of switches between any of the three main representations of the program (the code, the visualization, and the output). Figure 6 presents the results obtained by the eyetracker for the blurred (RFV-on) and fully focused (RFV-off) stimulus. On average, the experts performed 4.93 ($SD = 1.93$) and 8.86 ($SD = 4.09$) switches per minute under the RFV-on and RFV-off

conditions, respectively, whereas novices performed 7.00 ($SD = 2.13$) and 7.76 ($SD = 3.50$) switches per minute under the RFV-on and RFV-off conditions, respectively. The effect of RFV condition was significant [$F(1,16) = 7.82, p < .05$], and the interaction between level of experience and RFV condition was significant, $\alpha = .92$ [$F(1,16) = 3.59, p < .08$]. We observed a decrease in the number of switches per minute when the display was blurred, which was significant for experts [$t(7) = 2.53, p < .05$]. Moreover, the average number of switches per minute of novices was significantly correlated under the RFV-on and RFV-off conditions [$r(10) = .642, p = .046$], whereas the same correlation for experts was low and not significant [$r(8) = .068, p = .873$].

The mean fixation duration is often used as a measure of cognitive workload and depth of required processing (Goldberg & Kotval, 1998). We analyzed the mean fixation durations, measured by the eyetracker for the two experimental groups and the two blurring conditions, for the three main areas of the SDE interface and overall (Figure 7). The analysis of the results revealed an effect of RFV condition on mean fixation duration [$F(1,16) = 4.45, p < .051$], and no interaction between level of experience and RFV condition [$F(1,16) = 0.26, n.s.$]. The effect of experience on the mean fixation duration approached significance [$F(1,16) = 3.6, p = .076$]. The mean fixation durations on the three areas were significantly different [$F(2,16) = 10.13, p < .005$]. The planned paired t tests revealed that, for experts, the overall mean fixation duration and the mean fixation durations over the code AOI differed significantly between the RFV-on and RFV-off conditions [$t(7) = 2.80; t(7) = 2.66$, respectively; all $ps < .05$]. The overall mean fixation durations of the experts were 308.82 msec ($SD = 83.95$) and 263.09 msec ($SD = 70.60$) under the RFV-on and RFV-off conditions, respectively. For the code panel, the mean fixation durations of the expert group were 312.44 msec ($SD = 85.69$) and 268.23 msec ($SD = 73.64$) under the RFV-on and RFV-off conditions, respectively. For the novice group, there was no significant difference in the fixation durations between the RFV-on and RFV-off conditions, according to the pairwise

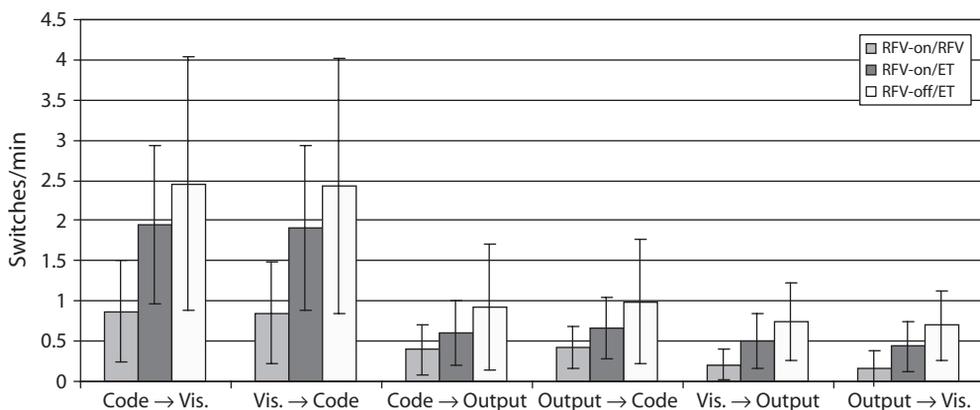


Figure 4. The average number of switches per minute as measured by the RFV and the eyetracker (ET) under the two conditions for all six types of switches.

Table 1
Correlations of Number of Switches per Minute Between RFV-On and RFV-Off Conditions, as Measured by the RFV and Eyetracker (ET), in the Format “Tool (Condition)”

| | Code→ Vis. | Vis.→ Code | Code→ Output | Output→ Code | Vis.→ Output | Output→ Vis. |
|-------------------------------|---------------|---------------|-----------------|-----------------|-----------------|-----------------|
| RFV (RFV-on) vs. ET (RFV-on) | .1694 | .1465 | .1741 | .4184 | .3716 | .0358 |
| ET (RFV-on) vs. ET (RFV-off) | .5578* | .5467* | .3027 | .2508 | -.3390 | -.0690 |
| RFV (RFV-on) vs. ET (RFV-off) | .0588 | .1579 | .0391 | -.0522 | -.1610 | -.1620 |

Note—Vis., visualization. * $p < .05$, two-tailed t test.

tests. The overall mean fixation durations of the novices were 396.90 msec ($SD = 112.29$) and 381.44 msec ($SD = 112.62$) under the RFV-on and RFV-off conditions, respectively. No statistically significant difference between them was found, and other measurements of novice mean fixation durations also did not differ significantly under the RFV-on and RFV-off conditions.

Summaries of the interviews. All of the experimental participants were informally interviewed after the debugging session, to investigate their attitudes and opinions about the display blurring. We discovered a general pattern occurring in the statements of subjective attitudes: Novice programmers greatly appreciated the fact that the environment reduced the wealth of visual stimuli, allowing them to concentrate better on extracting the information from the display. Experts, on the other hand, universally disliked the blurring feature of the environment. As an extreme example of negative attitude toward the display blurring, an expert—who declined to participate in the experiment shortly after the warm-up program appeared blurred on the screen—stated in the interview: “I do not want to work with that.” Consequently, no data were recorded for this person.

DISCUSSION

This study addressed an important issue: whether tracking of visual attention through eye-movement technology can be substituted by the tracking of a focused spot within an otherwise blurred stimulus. The eyetracker records a location of gaze direction, which is thought to be tightly coupled with the direction of visual attention. The RFV assumes that participants constantly look at the only focused

spot within otherwise blurred stimuli. We replicated one of the previous experiments that employed RFV technology (Romero, Lutz, et al., 2002), but we also recorded the gaze data, with the help of a remote eyetracker. We compared the data provided by these two tools and also analyzed the changes in the behavior of participants when the stimulus was presented in a blurred form. We presented several findings related to the performance and behavior data.

The RFV has already been validated previously (Jansen et al., 2003). However, the tasks used in the validation were not very complex and involved only one graphical representation of a problem, in contrast to the program debugging, which involved participants’ reasoning with multiple representations of the program. Moreover, the previous validation did not consider the possible effects of the blurring on the mental workload of the participants.

If participants’ performance is measured in terms of the number of bugs discovered under the two conditions, then no significant effect of display blurring was found on either of the experimental groups. Overall, the more experienced group performed better, but this was linked with their experience and was an expected result that has also been found by other empirical studies of novice and expert programmers (e.g., Gugerty & Olson, 1986). The performance of the more experienced participants, however, improved slightly under the blurring condition.

In comparison with the eyetracker, the RFV does not seem to accurately track the switching of visual attention of programmers while they are working with multiple representations of a program. The frequencies of visual attention switching as measured by the RFV were always lower in our experiment than were the frequencies measured by the remote eyetracker, and the two streams of data were

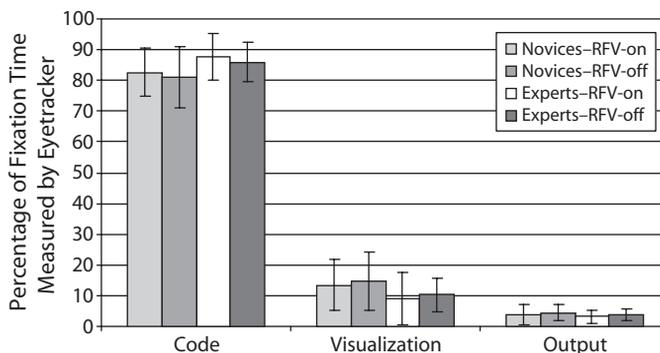


Figure 5. Proportions of accumulated fixation times spent on three areas of interest, as measured by the eyetracker.

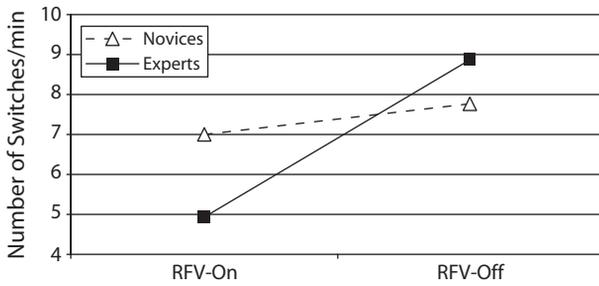


Figure 6. The average number of switches, as measured by the eyetracker.

not correlated under the restrictive blurred-display condition. Moreover, we found that the blurring condition interfered with the debugging strategies of the participants. Therefore, the results from the present experiment do not agree with the previous validation studies of RFV (Jansen et al., 2003). This finding can be explained both from a methodological view and from a theoretical perspective provided by previous studies of programmers.

When targets are reasonably large, gaze has been shown to provide faster ways of interaction than does mouse selection (Sibert & Jacob, 2000; Ware & Mikaelian, 1987). By observing the video recordings taken while the stimuli were blurred (for an example screen shot, see Figure 8), we revealed that it was often the case that participants gazed at locations other than the focused region as defined by the RFV position. Usually, the participants set the focused spot to a certain location and then performed several quick glances away and attention switches without moving the RFV-focused spot. In this way, the focused region was not used as the only location for extracting the information from the environment, but served as a kind of bookmark to the current representation. This behavior seemed to be frequent and intentional, most probably providing participants a memory refresh of the previously visited locations. In those situations, the costs of the required movements of the hand-controlled focus window were not justified, in comparison with the relatively effortless movement of the eyes. The RFV-based measures therefore cannot be equivalent to or estimates of the changes in visual attention, as is also shown by their low correlations to the eyetracking measures. When the display was not restricted, the cost

of a visual attention switch decreased further and was rewarded with unblurred information, as is evidenced by the significant increase in switching frequency.

To analyze where the differences between the RFV-on and RFV-off conditions come from, and whether the effects of the blurring interact with the levels of experience, we conducted a more detailed analysis using eye-movement data alone. The results reveal some effects of the blurring condition on gaze behavior. These effects were found to be more serious for more experienced participants, and they materialized on the attention-switching behavior rather than on the distribution of fixation times. Novice programmers used approximately the same strategy regardless of the display blurring, as can be seen from the high correlations in their switching frequency. Experts, on the other hand, when the display was blurred, changed their natural coordination strategies. When the information was easily available, under the RFV-off condition, experts seemed to integrate more information, as indicated by the increased number of switches.

The differences between the fixation durations of expert and novice participants were caused by the differences in levels of experience, and this result supports findings from other studies (e.g., Bednarik, Myller, Sutinen, & Tukiainen, 2005). Although we did not find any effect of the blurring on the fixation duration of novices, the increase in the fixation duration of experts was surprising, at first glance. However, as the experts changed their natural strategies and performed less information integration from the representations when the display was blurred, they had to spend more mental resources to complete the task. As a result, the increased fixation duration reflects the additional mental workload caused by the blurred display: Since experienced programmers are known to form good hypotheses about the comprehended problem and to be more able to remember specific parts of the code (and presumably also of the visualization and output), they had to decide at each moment exactly what information needed to be extracted from the restricted environment. This is in line with findings of Koenemann and Robertson (1991), who showed that experts concentrate only on relevant information while comprehending program code. Our results can be seen as extending this finding to multirepresentational programming environments: When restricted, experts conducted the coordination of the dif-

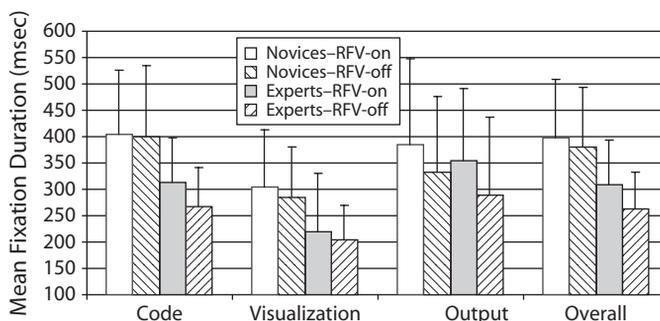


Figure 7. Mean fixation durations over three main areas and overall.

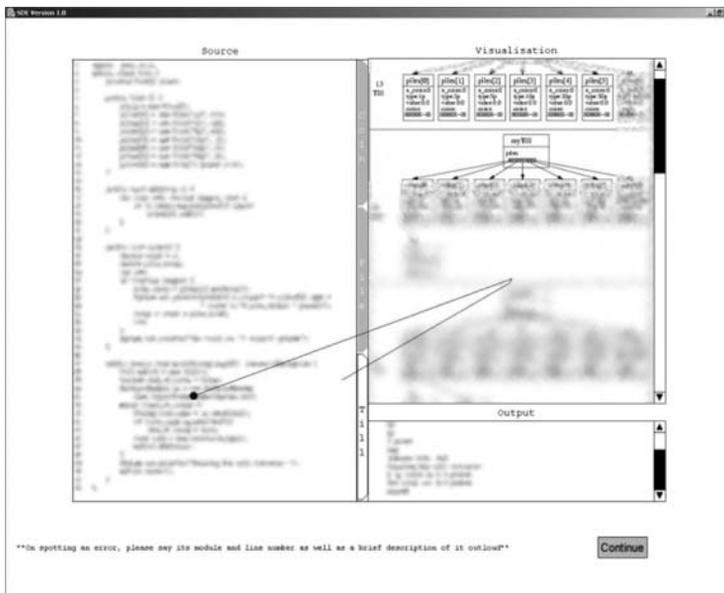


Figure 8. A screen shot from interaction, with 1 sec of gaze path and a fixation point superimposed. The focused region is on the top right.

ferent representations mentally rather than visually, yet with an equally good outcome.

We believe that the blurring of the display causes disturbances to the debugging behavior, and these disturbances were more serious for experts than for novices. The results of the interviews can be seen as additional support for this conclusion. Whereas the novices did not consider debugging with a blurring display an unusual task, experts complained about the unnaturalness of the task. However, the performance of the two groups was not influenced, suggesting that experts had to adopt some new, yet equally successful strategies to cope with the restricted view.

The RFV technique is cheap and requires no new equipment to record and analyze the experimental data, although some software efforts are required. However, it produces essentially the same stream of complex data as does eye-movement recording, so it does not simplify the analysis and interpretation of data. On the other hand, the stream of eye-movement data is often highly noisy, and is corrupted by the head movements of the participants or by additional reflections from the eyeglasses. For instance, for an accurate analysis of the switching frequency and total gaze durations, it is necessary that the gaze be constantly available. If the gaze is lost from the view of the eyetracking camera for a certain period, the computation of the eye-movement metrics cannot be accurate. Although eyetracking requires a calibration before the experiment, Jones and Mewhort (2004) have shown that the information filter (focus window) of the RFV (and similar devices) also needs to be calibrated.

Another inherent issue related to RFV-based studies concerns the question of experimental setting. For example, in the study we replicated, one might ask how natural the interaction and debugging strategies are when (1) the stimuli are precomputed, (2) the environment does not

allow for modifications of the source code, and (3) the participants have to use an unnatural interface. As seen from the results of this experiment, the blurring does indeed interfere with the natural strategies of participants. Jones and Mewhort (2004) suggested that the size and characteristics of the view in focus affect the information search strategies. In the present experiment, the size of the focused window was equal to those used in previous investigations using the RFV. If the size of the focused window and the level of blurring of the periphery were to be made more restrictive, to prevent the unwanted use of peripheral information, it is highly probable that the participants who rely on peripheral data would have to put more energy into moving the mouse to get the information they need. The resultant behavior would probably be even more disrupted, and the results of such a study would diverge further from natural conditions. However, this question will have to be answered in future studies.

Conclusions

Although the RFV is designed not to replace eyetrackers, but to be an alternative to the expensive technology, the results of this replication study show that some experimental tasks and designs might not fit well for the RFV. For those tasks, the visual attention data recorded using the RFV and eyetracking differ significantly. Moreover, the blurring of the stimuli changes the natural strategies used, depending on experience levels. We provided possible methodological and theoretical explanations of the differences found.

Although the RFV was introduced as a complementary tool for visual attention tracking, these findings make the use of the RFV, as well as the conclusions of the RFV-based experiments on complex problem-solving tasks, questionable.

AUTHOR NOTE

This work was supported by a grant from the Faculty of Science, University of Joensuu. Correspondence concerning this article should be addressed to R. Bednarik, Department of Computer Science, University of Joensuu, P.O. Box 111, 80101 Joensuu, Finland (e-mail: bednarik@cs.joensuu.fi).

REFERENCES

- BEDNARIK, R., MYLLER, N., SUTINEN, E., & TUKIAINEN, M. (2005). Effects of experience on gaze behavior during program animation. In P. Romero, J. Good, E. Acosta Chaparro, & S. Bryant (Eds.), *Proceedings of the 17th Annual Workshop of the Psychology of Programming Interest Group (PPIG '05)* (pp. 49-61). Brighton, U.K.: University of Sussex.
- BEDNARIK, R., & TUKIAINEN, M. (2004). Visual attention and representation switching in Java program debugging: A study using eye movement tracking. In E. Dunican & T. R. G. Green (Eds.), *Proceedings of the 16th Annual Workshop of the Psychology of Programming Interest Group (PPIG '04)* (pp. 159-169). Carlow, Ireland: Institute of Technology.
- FIX, V., WIEDENBECK, S., & SCHOLTZ, J. (1993). Mental representations of programs by novices and experts. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '93)* (pp. 74-79). New York: ACM Press.
- FUTRELLE, R. P., & RUMSHISKY, A. (2001). Discourse structure of text-graphics documents. *Proceedings of the 1st International Symposium on Smart Graphics*. Hawthorne, New York: ACM Press.
- GOLDBERG, J. H., & KOTVAL, X. P. (1998). Eye movement-based evaluation of the computer interface. In S. K. Kumar (Ed.), *Advances in occupational ergonomics and safety* (pp. 529-532). Amsterdam: IOS Press.
- GUGERTY, L., & OLSON, G. M. (1986). Comprehension differences in debugging by skilled and novice programmers. In E. Soloway & S. Iyengar (Eds.), *Empirical studies of programmers: First workshop* (pp. 13-27). Norwood, NJ: Ablex.
- HYÖNÄ, J., LORCH, R. F., JR., & KAAKINEN, J. K. (2002). Individual differences in reading to summarize expository text: Evidence from eye fixation patterns. *Journal of Educational Psychology*, *94*, 44-55.
- JANSEN, A. R., BLACKWELL, A. F., & MARRIOTT, K. (2003). A tool for tracking visual attention: The Restricted Focus Viewer. *Behavior Research Methods, Instruments, & Computers*, *35*, 57-69.
- JONES, M. N., & MEWHORT, D. J. K. (2004). Tracking attention with the focus-window technique: The information filter must be calibrated. *Behavior Research Methods, Instruments, & Computers*, *36*, 270-276.
- JUST, M. A., & CARPENTER, P. A. (1976). Eye fixations and cognitive processes. *Cognitive Psychology*, *8*, 441-480.
- KOENEMANN, J., & ROBERTSON, S. P. (1991). Expert problem solving strategies for program comprehension. In S. P. Robertson, G. M. Olson, & J. S. Olson (Eds.), *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Reaching through technology* (pp. 125-130). New York: ACM Press.
- LAW, B., ATKINS, M. S., KIRKPATRICK, A. E., & LOMAX, A. J. (2004). Eye gaze patterns differentiate novice and experts in a virtual laparoscopic surgery training environment. *Proceedings of the 2004 Symposium on Eye Tracking Research and Applications* (pp. 41-48). New York: ACM Press.
- RAYNER, K. (1998). Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin*, *124*, 372-422.
- ROMERO, P., COX, R., DU BOULAY, B., & LUTZ, R. (2002). Visual attention and representation switching during Java program debugging: A study using the Restricted Focus Viewer. In *Diagrammatic Representation and Inference: Second International Conference, Diagrams 2002, Callaway Gardens, GA, USA, April 18-20, 2002: Proceedings* (Lecture Notes in Computer Science, Vol. 2317, pp. 221-235). Berlin: Springer.
- ROMERO, P., DU BOULAY, B., COX, R., & LUTZ, R. (2003). Java debugging strategies in multi-representational environments. In M. Petre (Ed.), *Proceedings of the 15th Annual Workshop of the Psychology of Programming Interest Group (PPIG '03)* (pp. 421-434).
- ROMERO, P., DU BOULAY, B., LUTZ, R., & COX, R. (2003). The effects of graphical and textual visualisations in multi-representational debugging environments. In J. Hosking & P. Cox (Eds.), *Proceedings of the 2003 IEEE Symposia on Human Centric Computing Languages and Environments* (pp. 236-238). Piscataway, NJ: IEEE Computer Society.
- ROMERO, P., LUTZ, R., COX, R., & DU BOULAY, B. (2002). Coordination of multiple external representations during Java program debugging. In S. Wiedenbeck & M. Petre (Eds.), *Proceedings of the 2002 IEEE Symposia on Human Centric Computing Languages and Environments* (pp. 207-214). Piscataway, NJ: IEEE Computer Society.
- STIBERT, L. E., & JACOB, R. J. K. (2000). Evaluation of eye gaze interaction. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 281-288). New York: ACM Press.
- TARASEWICH, P., & FILLION, S. (2004). Discount eye tracking: The Enhanced Restricted Focus Viewer. *Proceedings of the 10th Americas Conference on Information Systems* (pp. 1-9). New York: AMCIS.
- VESSEY, I. (1985). Expertise in debugging computer programs: A process analysis. *International Journal of Man-Machine Studies*, *23*, 459-494.
- WARE, C., & MIKAEILIAN, H. H. (1987). An evaluation of an eye tracker as a device for computer input. *Proceedings of the SIGCHI/GI Conference on Human Factors in Computing Systems and Graphics Interface (CHI '87)* (pp. 183-188). New York: ACM Press.

(Manuscript received June 16, 2005;
revision accepted for publication January 21, 2006.)